UCONN
UNIVERSITY OF CONNECTICUT

Global Optimization of Advanced Formulations with EAGO.jl: Recent Advances

Matthew D. Stuber, PhD, Assistant Professor

stuber@alum.mit.edu

Robert X. Gottlieb, PhD Student

Matthew E. Wilhelm, PhD Candidate

3-6 JULY
EURO 2022

AALTO UNIVERSITY
ESPOO FINLAND

Process Systems and Operations Research Laboratory

Key Contributors

Matthew E. Wilhelm
PhD Candidate
Original EAGO Developer

Robert X. Gottlieb
PhD Student

PSOR Lab

Dept. of Chemical and Biomolecular Engineering

University of Connecticut

Outline

- Background and Motivation
  - Get EAGO
  - “Advanced formulations”
  - Deterministic global optimization
- EAGO.jl Core Features
  - B&B, McCormick
- Recent Advances
  - ANNs
  - Global dynamic optimization and GPU acceleration
- Conclusion

Background: EAGO/Julia

How do you get EAGO?

From Julia package manager:

```
(@v1.7) pkg> add EAGO
```

```
julia> using Pkg;

julia> Pkg.add("EAGO")
```

From GitHub:

https://www.github.com/PSORLab/EAGO.jl

Background: EAGO/Julia

How do you get EAGO?

From Julia package manager:

```
(@v1.7) pkg> add EAGO
```

```
julia> using Pkg;

julia> Pkg.add("EAGO")
```

From GitHub:

https://www.github.com/PSORLab/EAGO.jl

How do you use EAGO?

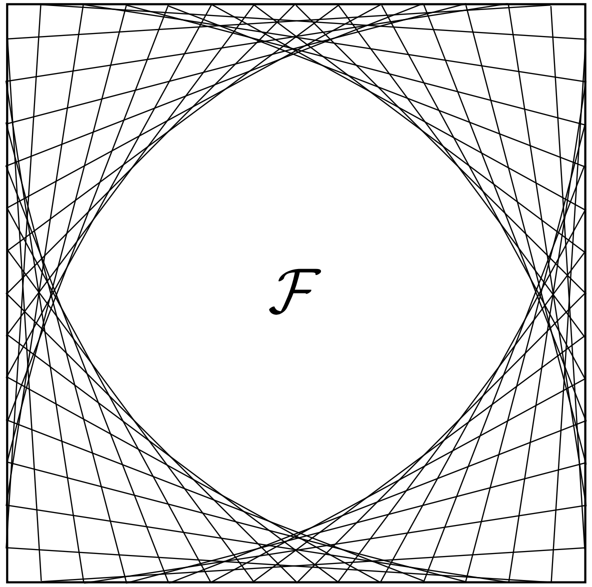
As a solver in the open-source algebraic modeling language JuMP.

As a stand-alone solver.

Background: Advanced Formulations

- May be parametric and/or have multilevel structure

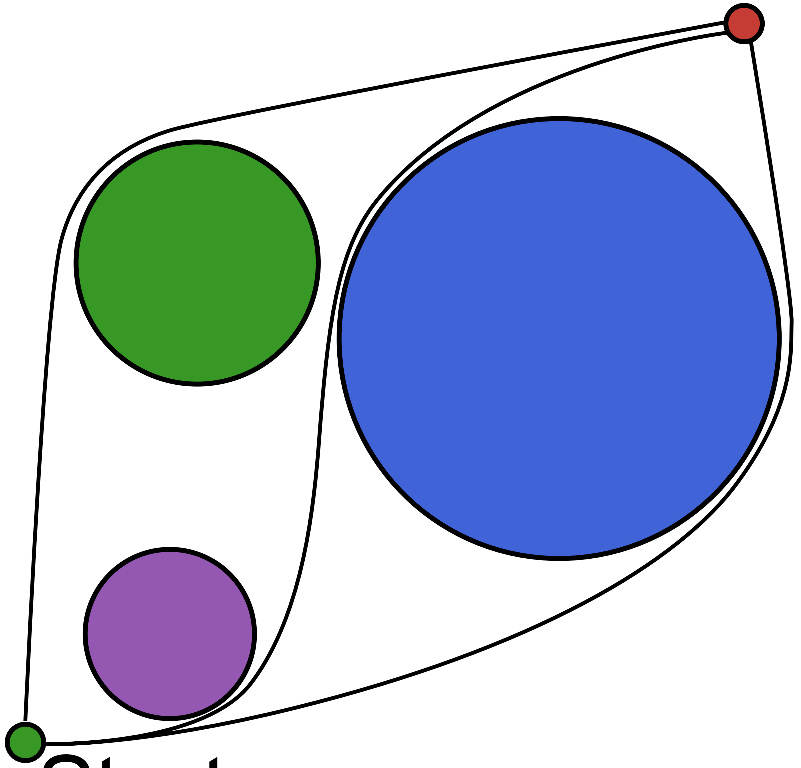
$$\begin{aligned}\min_{\mathbf{x}\in X} & \; f(\mathbf{x}) \\ \text{s.t.} & \; \mathbf{g}(\mathbf{x},\mathbf{p}) \leq \mathbf{0}, \forall \mathbf{p} \in P \\ & \; \text{card}(P) = \infty\end{aligned}$$



$$\begin{aligned}\min_{\mathbf{x},\mathbf{y}} & \; f(\mathbf{x},\mathbf{y}) \\ \text{s.t.} & \; g(\mathbf{x},\mathbf{y}) \leq 0 \\ & \; \mathbf{x} \in X \in \mathbb{R}^{n_x} \\ & \; \mathbf{y} \in \arg\max_{\mathbf{z}\in Y(\mathbf{x})} h(\mathbf{x},\mathbf{z})\end{aligned}$$

Background: Advanced Formulations

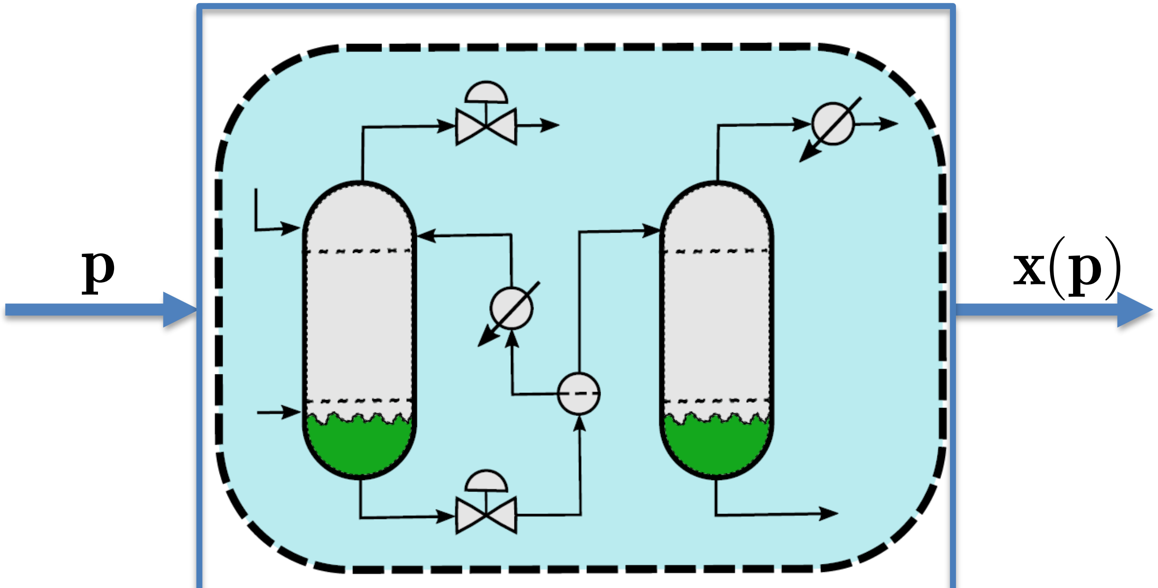
- May involve differential equations (ODEs, PDEs, DAEs)



$$\begin{aligned} \phi^* = & \min_{\mathbf{p} \in P \subset \mathbb{R}^{n_p}} \phi(\mathbf{x}(\mathbf{p}, t_f), \mathbf{p}) \\ \text{s.t.} \;\; & \dot{\mathbf{x}}(\mathbf{p}, t) = \mathbf{f}(\mathbf{x}(\mathbf{p}, t), \mathbf{p}, t), \forall t \in I = [t_0, t_f] \\ & \mathbf{x}(\mathbf{p}, t_0) = \mathbf{x}_0(\mathbf{p}) \\ & \mathbf{g}(\mathbf{x}(\mathbf{p}, t_f), \mathbf{p}) \leq \mathbf{0} \end{aligned}$$

Background: Advanced Formulations

- May involve "white box" simulations



$$\mathbf{h}(\hat{\mathbf{x}}, \mathbf{p}) = \mathbf{0}$$

$$\begin{aligned} f^* = \min_{\mathbf{p} \in P} & \; f\Big(\mathbf{x}(\mathbf{p})\Big) \\ \text{s.t.} & \; \mathbf{g}\Big(\mathbf{x}(\mathbf{p})\Big) \leq \mathbf{0} \\ & \; \mathbf{x}(\mathbf{p}) : \mathbf{h}\Big(\mathbf{x}(\mathbf{p}), \mathbf{p}\Big) = \mathbf{0} \end{aligned}$$

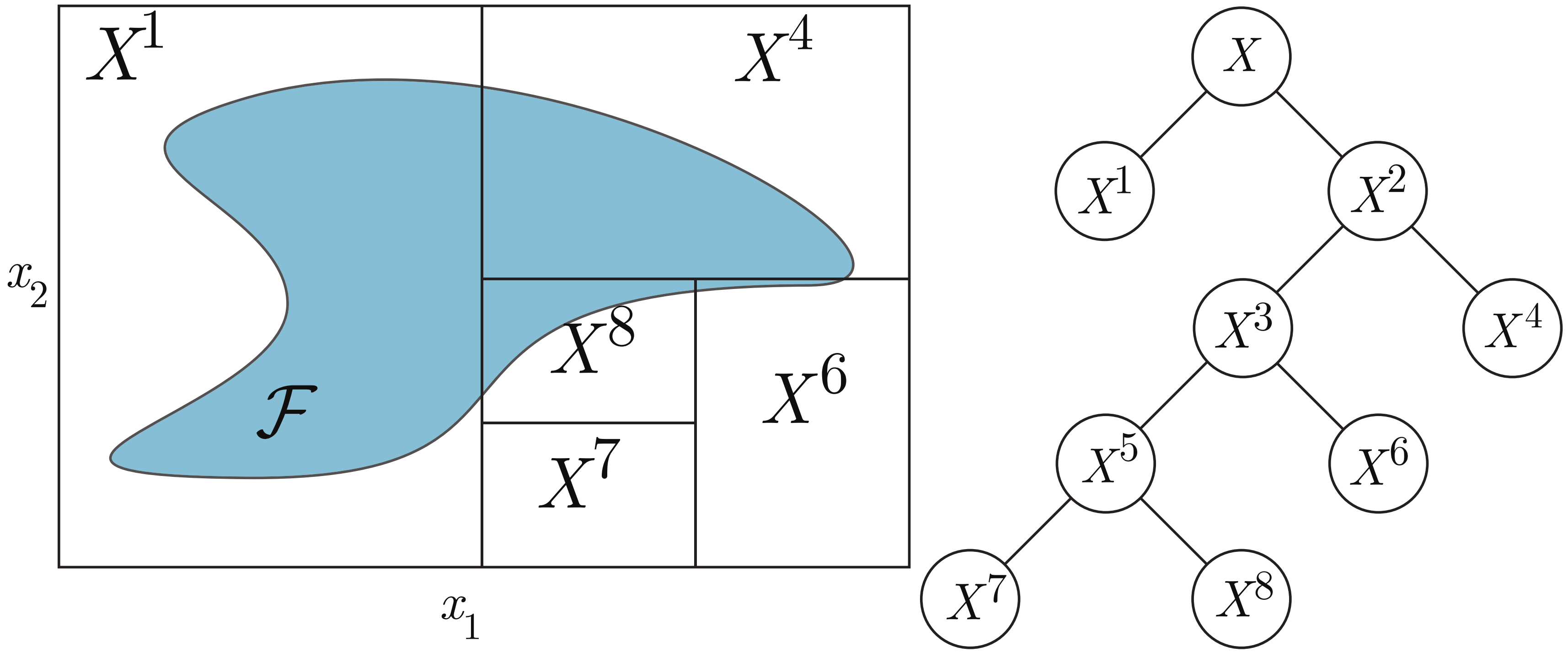
Background: Advanced Formulations

- May be parametric and/or multilevel structure
- May involve differential equations (ODEs, PDEs)
- May involve "white box" simulations

$$\begin{aligned} \min_{var} \; & objective \\ \text{s.t.} \; & advanced \\ & \; algorithms \end{aligned}$$

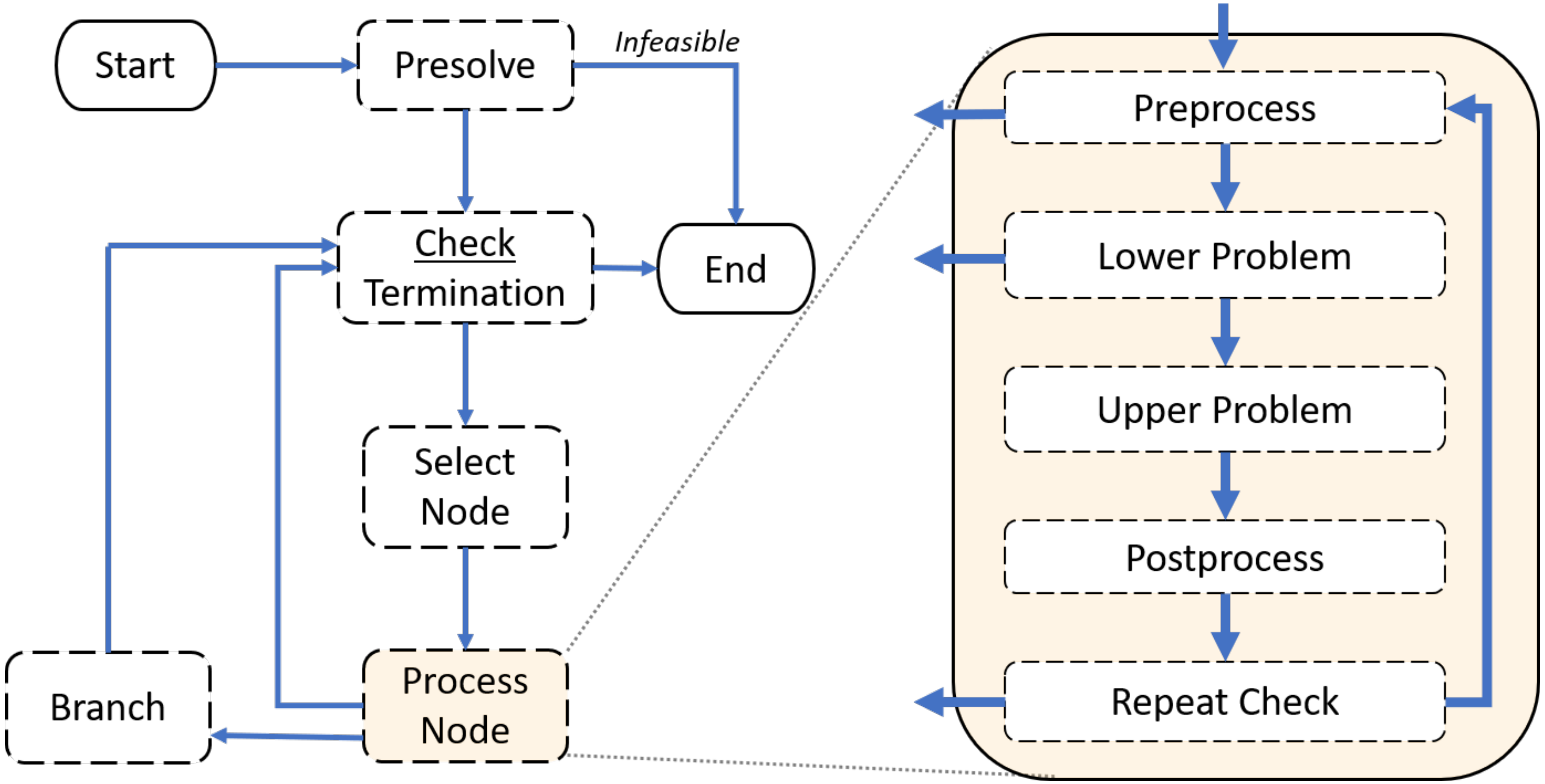
Background: Deterministic Global Optimization

- Deterministic search: branch-and-bound



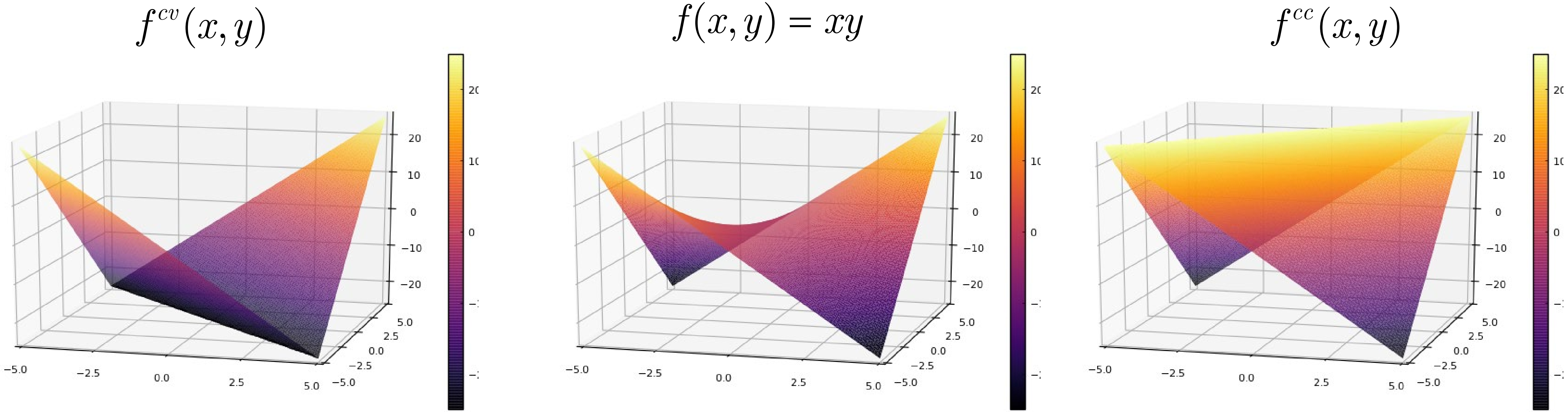
EAGO: Core

- Customizable and feature-rich branch-and-bound
  - McCormick-based convex/concave relaxations



EAGO: McCormick Relaxations

- Most broadly known for convex/concave relaxations of bilinear terms



EAGO: McCormick Relaxations

Relaxations of g(x) at x in X

⋮

Relaxations of h(x) at x in X

Apply **f** *composite relaxation rules*

$$\mathbf{y} = \mathbf{f}(\mathbf{g}(\mathbf{x}), \ldots, \mathbf{h}(\mathbf{x}))$$

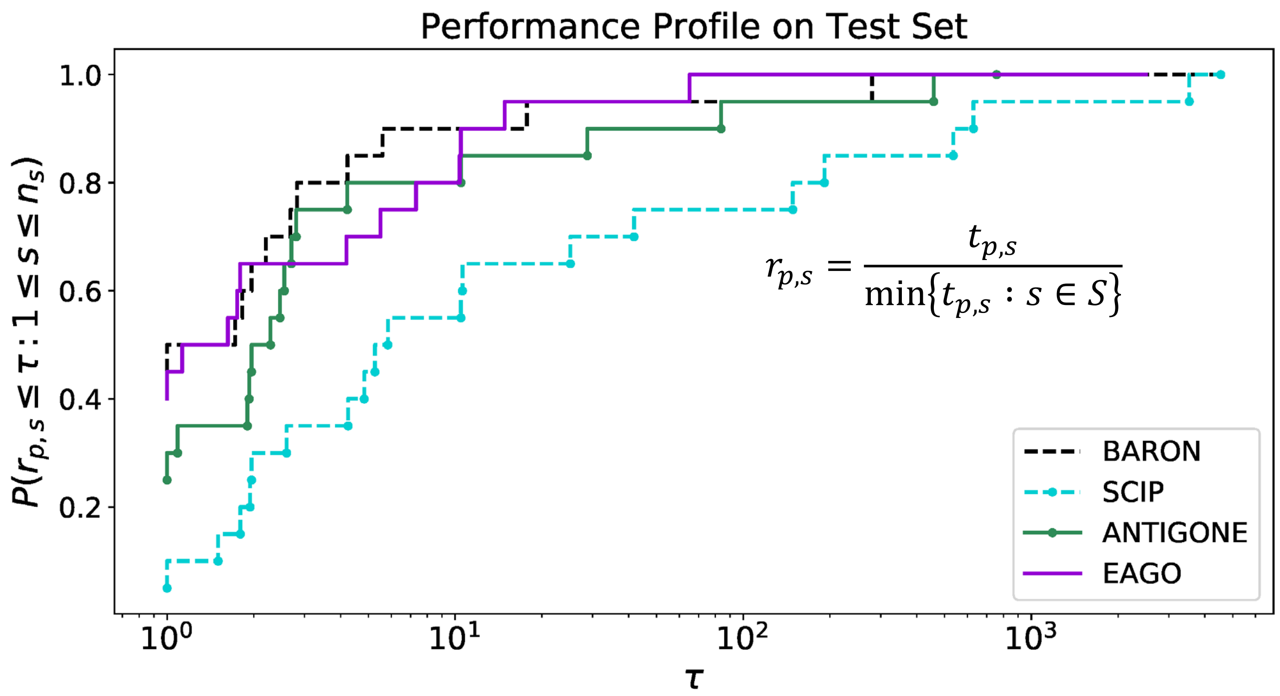
Relaxations of f(x) at x in X

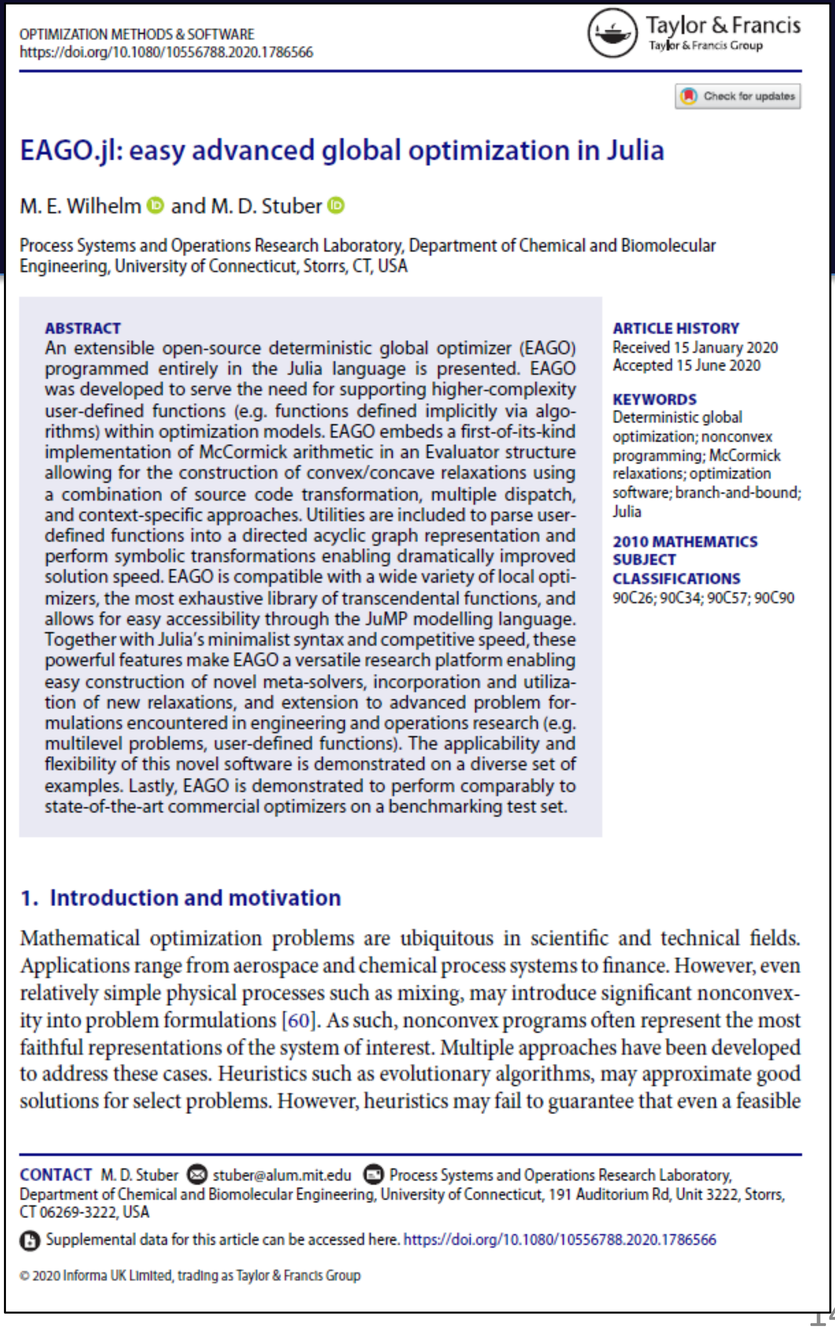
- Improved (tighter) relaxations of composite bilinear terms*
- Supports a variety of nonlinear expressions:
  - **Common algebraic expressions:** *log, log2, log10, exp, exp2, exp10, sqrt, +, -, ^, min, max, /,* x, *abs, step, cbrt, …*
  - **Trigonometric Functions:** *sin, cos, tan, asin, acos, atan, sec, csc, cot, asec, acsc, acot…*
  - **Hyperbolic Functions:** *sinh, cosh, tanh, asinh, acosh, atanh, sech, csch, coth, acsch, acoth*
  - **Special Functions:** *erf, erfc, erfinv, erfcinv*
  - **Activation Functions*:** *relu, leaky_relu, sigmoid, softsign, softplus, maxtanh, gelu, elu, selu, silu, …*
  - ***Common Algebraic Expressions:*** *xlogx, arh, xexpax*

**Under Review*

Published Results

- EAGO exhibits competitive performance on benchmarking set





OPTIMIZATION METHODS & SOFTWARE
https://doi.org/10.1080/10556788.2020.1786566
Taylor & Francis
Taylor & Francis Group
Check for updates

EAGO.jl: easy advanced global optimization in Julia

M. E. Wilhelm and M. D. Stuber

Process Systems and Operations Research Laboratory, Department of Chemical and Biomolecular Engineering, University of Connecticut, Storrs, CT, USA

ABSTRACT

An extensible open-source deterministic global optimizer (EAGO) programmed entirely in the Julia language is presented. EAGO was developed to serve the need for supporting higher-complexity user-defined functions (e.g. functions defined implicitly via algorithms) within optimization models. EAGO embeds a first-of-its-kind implementation of McCormick arithmetic in an Evaluator structure allowing for the construction of convex/concave relaxations using a combination of source code transformation, multiple dispatch, and context-specific approaches. Utilities are included to parse user-defined functions into a directed acyclic graph representation and perform symbolic transformations enabling dramatically improved solution speed. EAGO is compatible with a wide variety of local optimizers, the most exhaustive library of transcendental functions, and allows for easy accessibility through the JuMP modelling language. Together with Julia's minimalist syntax and competitive speed, these powerful features make EAGO a versatile research platform enabling easy construction of novel meta-solvers, incorporation and utilization of new relaxations, and extension to advanced problem formulations encountered in engineering and operations research (e.g. multilevel problems, user-defined functions). The applicability and flexibility of this novel software is demonstrated on a diverse set of examples. Lastly, EAGO is demonstrated to perform comparably to state-of-the-art commercial optimizers on a benchmarking test set.

ARTICLE HISTORY
Received 15 January 2020
Accepted 15 June 2020

KEYWORDS
Deterministic global optimization; nonconvex programming; McCormick relaxations; optimization software; branch-and-bound; Julia

2010 MATHEMATICS SUBJECT CLASSIFICATIONS
90C26; 90C34; 90C57; 90C90

1. Introduction and motivation

Mathematical optimization problems are ubiquitous in scientific and technical fields. Applications range from aerospace and chemical process systems to finance. However, even relatively simple physical processes such as mixing, may introduce significant nonconvexity into problem formulations [60]. As such, nonconvex programs often represent the most faithful representations of the system of interest. Multiple approaches have been developed to address these cases. Heuristics such as evolutionary algorithms, may approximate good solutions for select problems. However, heuristics may fail to guarantee that even a feasible

CONTACT M. D. Stuber stuber@alum.mit.edu Process Systems and Operations Research Laboratory, Department of Chemical and Biomolecular Engineering, University of Connecticut, 191 Auditorium Rd, Unit 3222, Storrs, CT 06269-3222, USA

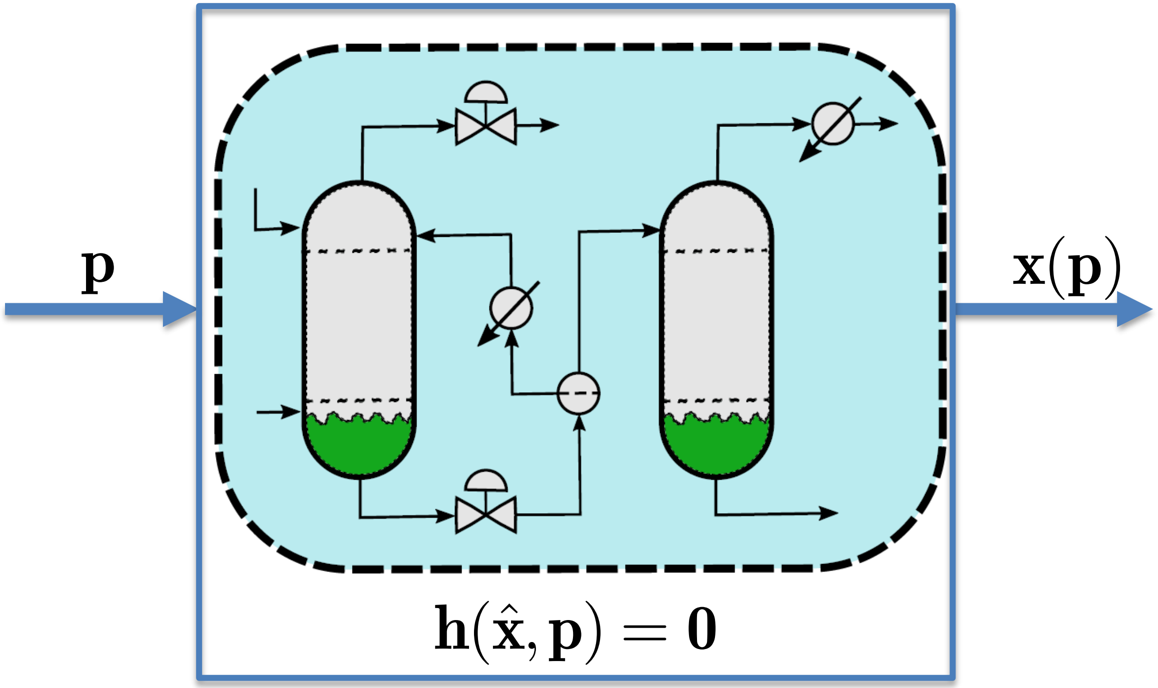
Supplemental data for this article can be accessed here. https://doi.org/10.1080/10556788.2020.1786566

Recent Advances

Recent Advances: ANNs

- Optimization of ANNs

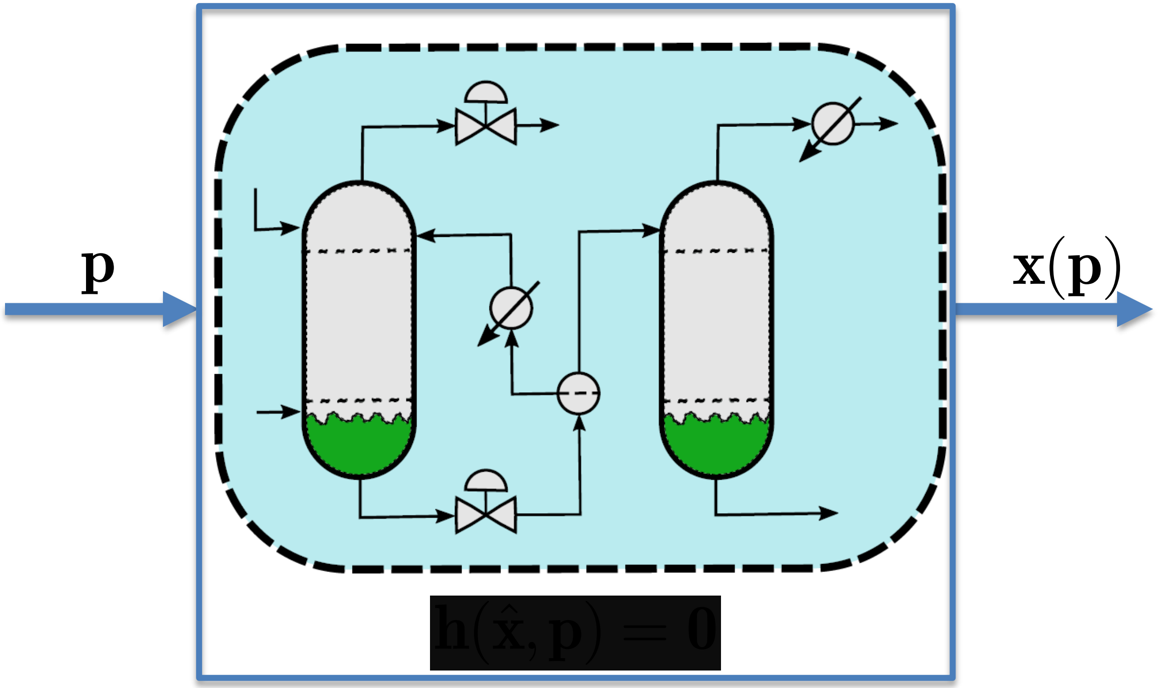
$$\min_{\mathbf{p}\in P} \phi(\mathbf{p}, \mathbf{x}(\mathbf{p}))$$
$$\text{s.t.} \quad \mathbf{g}(\mathbf{p}, \mathbf{x}(\mathbf{p})) \leq \mathbf{0}$$



Recent Advances: ANNs

- Optimization of ANNs

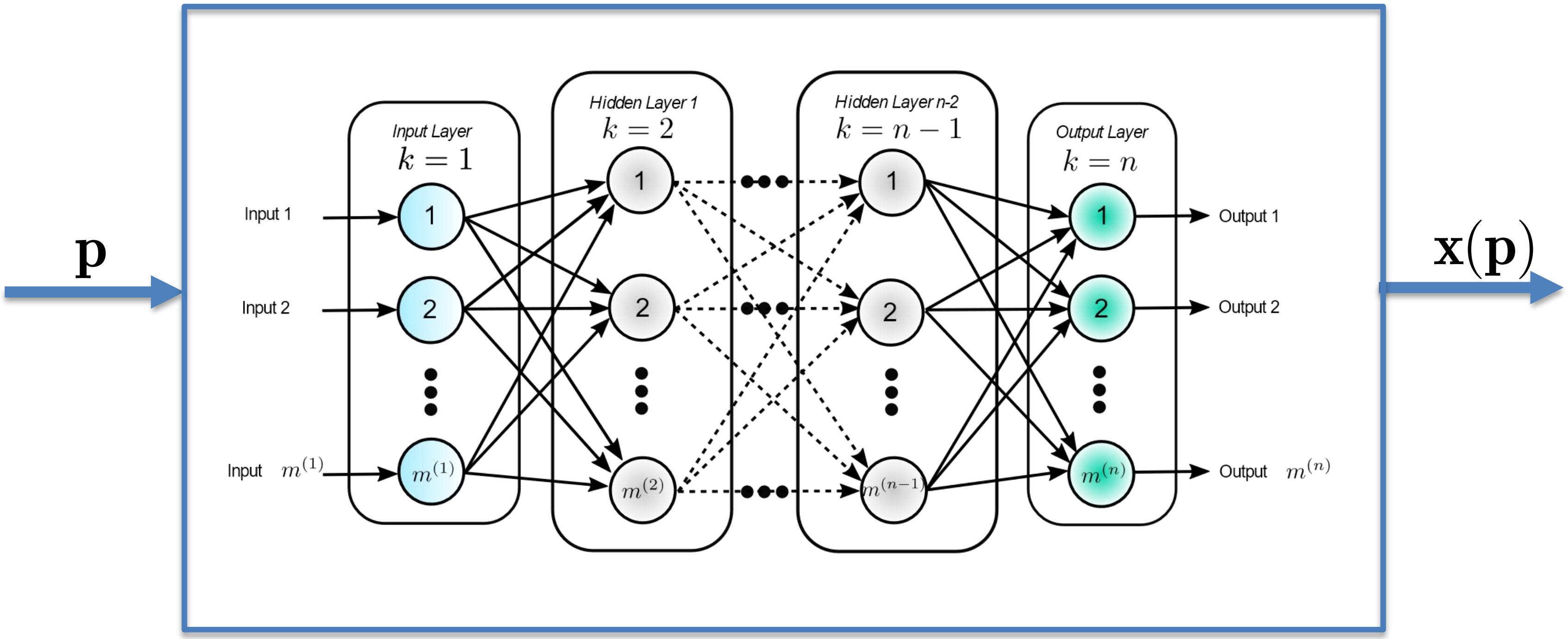
$$\min_{\mathbf{p}\in P} \phi(\mathbf{p}, \mathbf{x}(\mathbf{p}))$$
$$\text{s.t.} \;\; \mathbf{g}(\mathbf{p}, \mathbf{x}(\mathbf{p})) \le \mathbf{0}$$



Recent Advances: ANNs

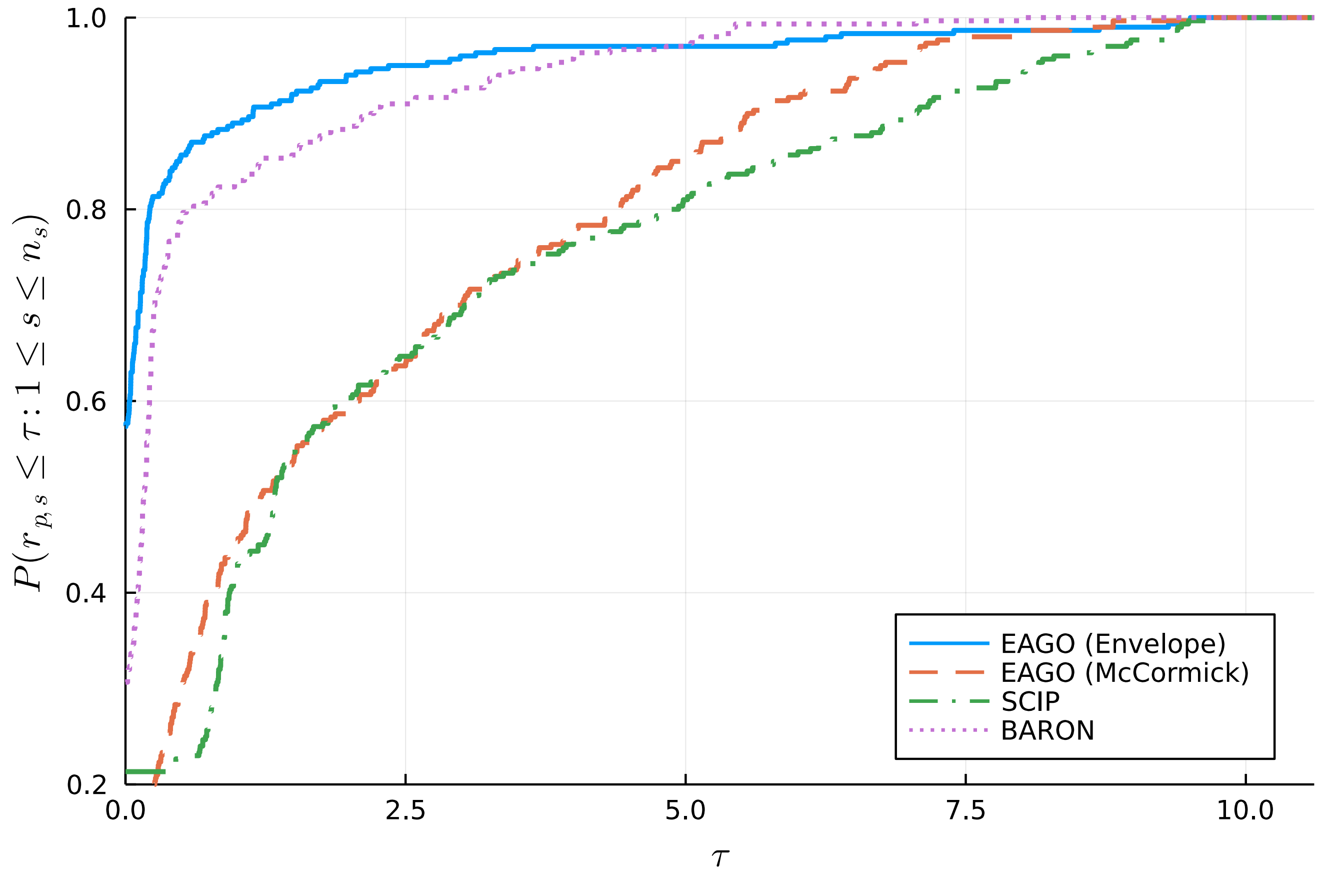
- Optimization of ANNs

$$\min_{\mathbf{p}\in P} \phi(\mathbf{p}, \mathbf{x}(\mathbf{p}))$$
$$\text{s.t. } \mathbf{g}(\mathbf{p}, \mathbf{x}(\mathbf{p})) \leq \mathbf{0}$$



Recent Advances: ANNs

- Optimization of ANNs*

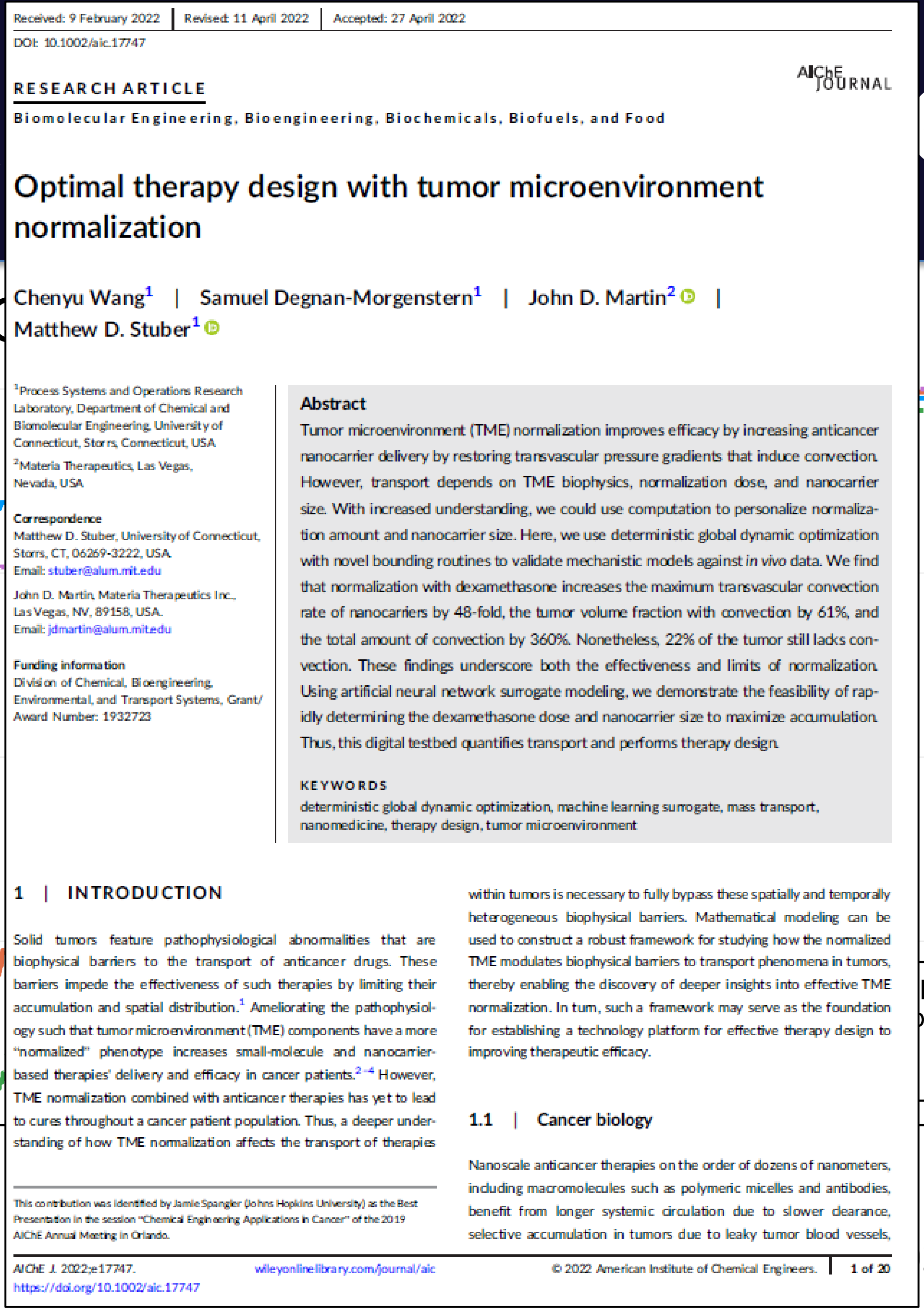


Solver	Solved	Unsolved
EAGO (Envelope)	280 (93.3%)	20 (6.7%)
EAGO (Naïve McCormick)	260 (86.7%)	40 (13.3%)
SCIP	240 (80.0%)	60 (20.0%)
BARON	273 (91.0%)	27 (9.0%)

**Under Review*

R… es:

1.0
0.8
0.6
0.4
0.2
0.0
10.0
$P(r_{p,s} \leq \tau : 1 \leq s \leq n_s)$
lope)
rmick)



Received: 9 February 2022 | Revised: 11 April 2022 | Accepted: 27 April 2022

DOI: 10.1002/aic.17747

AIChE JOURNAL

RESEARCH ARTICLE

Biomolecular Engineering, Bioengineering, Biochemicals, Biofuels, and Food

Optimal therapy design with tumor microenvironment normalization

Chenyu Wang[1] | Samuel Degnan-Morgenstern[1] | John D. Martin[2] | Matthew D. Stuber[1]

[1]Process Systems and Operations Research Laboratory, Department of Chemical and Biomolecular Engineering, University of Connecticut, Storrs, Connecticut, USA

[2]Materia Therapeutics, Las Vegas, Nevada, USA

Correspondence

Matthew D. Stuber, University of Connecticut, Storrs, CT, 06269-3222, USA.
Email: stuber@alum.mit.edu

John D. Martin, Materia Therapeutics Inc., Las Vegas, NV, 89158, USA.
Email: jdmartin@alum.mit.edu

Funding information

Division of Chemical, Bioengineering, Environmental, and Transport Systems, Grant/Award Number: 1932723

Abstract

Tumor microenvironment (TME) normalization improves efficacy by increasing anticancer nanocarrier delivery by restoring transvascular pressure gradients that induce convection. However, transport depends on TME biophysics, normalization dose, and nanocarrier size. With increased understanding, we could use computation to personalize normalization amount and nanocarrier size. Here, we use deterministic global dynamic optimization with novel bounding routines to validate mechanistic models against *in vivo* data. We find that normalization with dexamethasone increases the maximum transvascular convection rate of nanocarriers by 48-fold, the tumor volume fraction with convection by 61%, and the total amount of convection by 360%. Nonetheless, 22% of the tumor still lacks convection. These findings underscore both the effectiveness and limits of normalization. Using artificial neural network surrogate modeling, we demonstrate the feasibility of rapidly determining the dexamethasone dose and nanocarrier size to maximize accumulation. Thus, this digital testbed quantifies transport and performs therapy design.

KEYWORDS

deterministic global dynamic optimization, machine learning surrogate, mass transport, nanomedicine, therapy design, tumor microenvironment

1 | INTRODUCTION

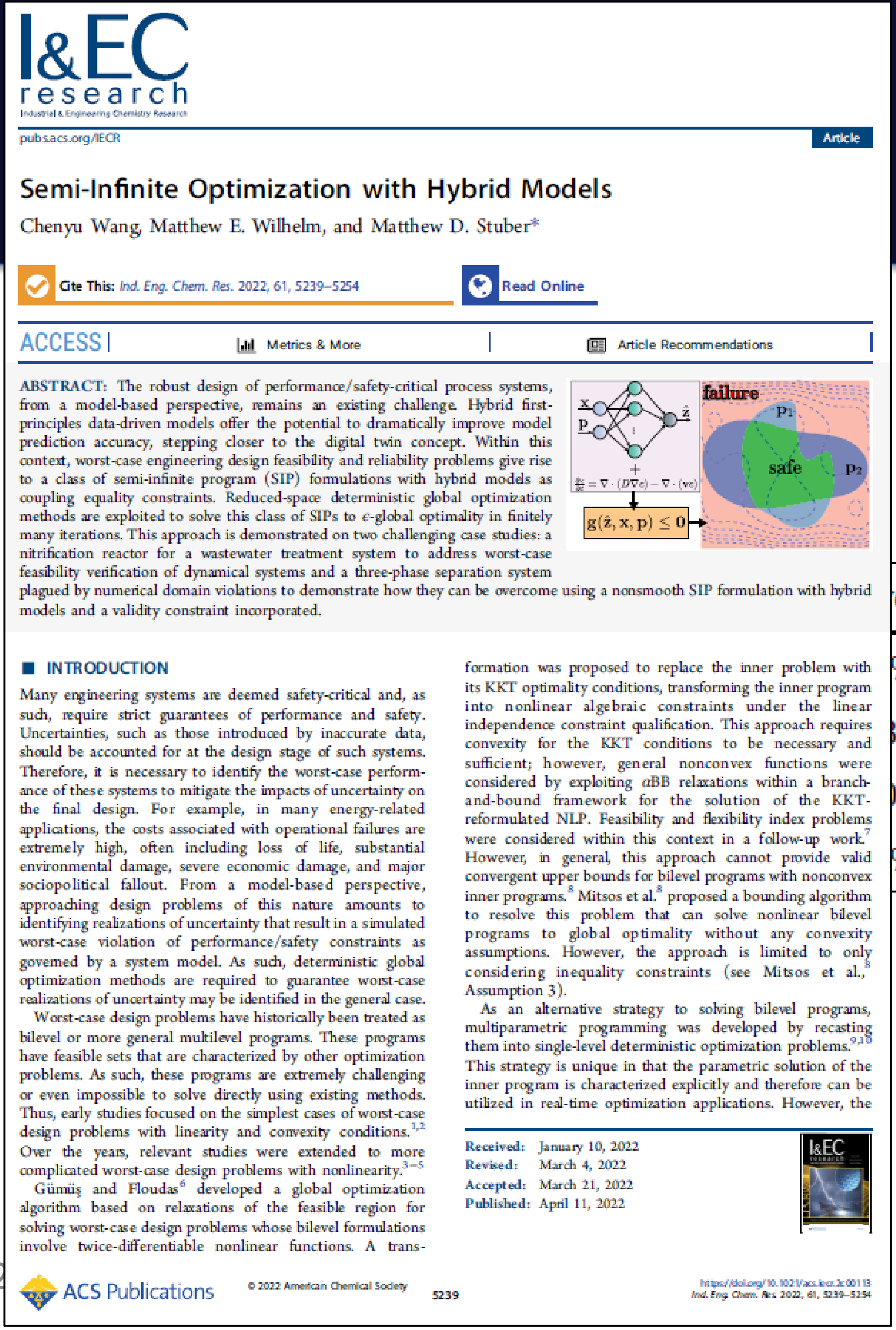
Solid tumors feature pathophysiological abnormalities that are biophysical barriers to the transport of anticancer drugs. These barriers impede the effectiveness of such therapies by limiting their accumulation and spatial distribution.[1] Ameliorating the pathophysiology such that tumor microenvironment (TME) components have a more "normalized" phenotype increases small-molecule and nanocarrier-based therapies' delivery and efficacy in cancer patients.[2–4] However, TME normalization combined with anticancer therapies has yet to lead to cures throughout a cancer patient population. Thus, a deeper understanding of how TME normalization affects the transport of therapies within tumors is necessary to fully bypass these spatially and temporally heterogeneous biophysical barriers. Mathematical modeling can be used to construct a robust framework for studying how the normalized TME modulates biophysical barriers to transport phenomena in tumors, thereby enabling the discovery of deeper insights into effective TME normalization. In turn, such a framework may serve as the foundation for establishing a technology platform for effective therapy design to improving therapeutic efficacy.

1.1 | Cancer biology

Nanoscale anticancer therapies on the order of dozens of nanometers, including macromolecules such as polymeric micelles and antibodies, benefit from longer systemic circulation due to slower clearance, selective accumulation in tumors due to leaky tumor blood vessels,

This contribution was identified by Jamie Spangler (Johns Hopkins University) as the Best Presentation in the session "Chemical Engineering Applications in Cancer" of the 2019 AIChE Annual Meeting in Orlando.

https://doi.org/10.1002/aic.17747



I&EC research

Industrial & Engineering Chemistry Research

pubs.acs.org/IECR Article

Semi-Infinite Optimization with Hybrid Models

Chenyu Wang, Matthew E. Wilhelm, and Matthew D. Stuber*

Cite This: *Ind. Eng. Chem. Res.* 2022, 61, 5239−5254 Read Online

ACCESS | Metrics & More | Article Recommendations

ABSTRACT: The robust design of performance/safety-critical process systems, from a model-based perspective, remains an existing challenge. Hybrid first-principles data-driven models offer the potential to dramatically improve model prediction accuracy, stepping closer to the digital twin concept. Within this context, worst-case engineering design feasibility and reliability problems give rise to a class of semi-infinite program (SIP) formulations with hybrid models as coupling equality constraints. Reduced-space deterministic global optimization methods are exploited to solve this class of SIPs to ϵ-global optimality in finitely many iterations. This approach is demonstrated on two challenging case studies: a nitrification reactor for a wastewater treatment system to address worst-case feasibility verification of dynamical systems and a three-phase separation system plagued by numerical domain violations to demonstrate how they can be overcome using a nonsmooth SIP formulation with hybrid models and a validity constraint incorporated.

INTRODUCTION

Many engineering systems are deemed safety-critical and, as such, require strict guarantees of performance and safety. Uncertainties, such as those introduced by inaccurate data, should be accounted for at the design stage of such systems. Therefore, it is necessary to identify the worst-case performance of these systems to mitigate the impacts of uncertainty on the final design. For example, in many energy-related applications, the costs associated with operational failures are extremely high, often including loss of life, substantial environmental damage, severe economic damage, and major sociopolitical fallout. From a model-based perspective, approaching design problems of this nature amounts to identifying realizations of uncertainty that result in a simulated worst-case violation of performance/safety constraints as governed by a system model. As such, deterministic global optimization methods are required to guarantee worst-case realizations of uncertainty may be identified in the general case.

Worst-case design problems have historically been treated as bilevel or more general multilevel programs. These programs have feasible sets that are characterized by other optimization problems. As such, these programs are extremely challenging or even impossible to solve directly using existing methods. Thus, early studies focused on the simplest cases of worst-case design problems with linearity and convexity conditions.[1,2] Over the years, relevant studies were extended to more complicated worst-case design problems with nonlinearity.[3−5]

Gümüş and Floudas[6] developed a global optimization algorithm based on relaxations of the feasible region for solving worst-case design problems whose bilevel formulations involve twice-differentiable nonlinear functions. A transformation was proposed to replace the inner problem with its KKT optimality conditions, transforming the inner program into nonlinear algebraic constraints under the linear independence constraint qualification. This approach requires convexity for the KKT conditions to be necessary and sufficient; however, general nonconvex functions were considered by exploiting αBB relaxations within a branch-and-bound framework for the solution of the KKT-reformulated NLP. Feasibility and flexibility index problems were considered within this context in a follow-up work.[7] However, in general, this approach cannot provide valid convergent upper bounds for bilevel programs with nonconvex inner programs.[8] Mitsos et al.[8] proposed a bounding algorithm to resolve this problem that can solve nonlinear bilevel programs to global optimality without any convexity assumptions. However, the approach is limited to only considering inequality constraints (see Mitsos et al.,[8] Assumption 3).

As an alternative strategy to solving bilevel programs, multiparametric programming was developed by recasting them into single-level deterministic optimization problems.[9,10] This strategy is unique in that the parametric solution of the inner program is characterized explicitly and therefore can be utilized in real-time optimization applications. However, the

Received: January 10, 2022
Revised: March 4, 2022
Accepted: March 21, 2022
Published: April 11, 2022

https://doi.org/10.1021/acs.iecr.2c00113
Ind. Eng. Chem. Res. 2022, 61, 5239−5254

ed
%)
3%)
0%)
%)

Recent Advances: ODEs/DAEs

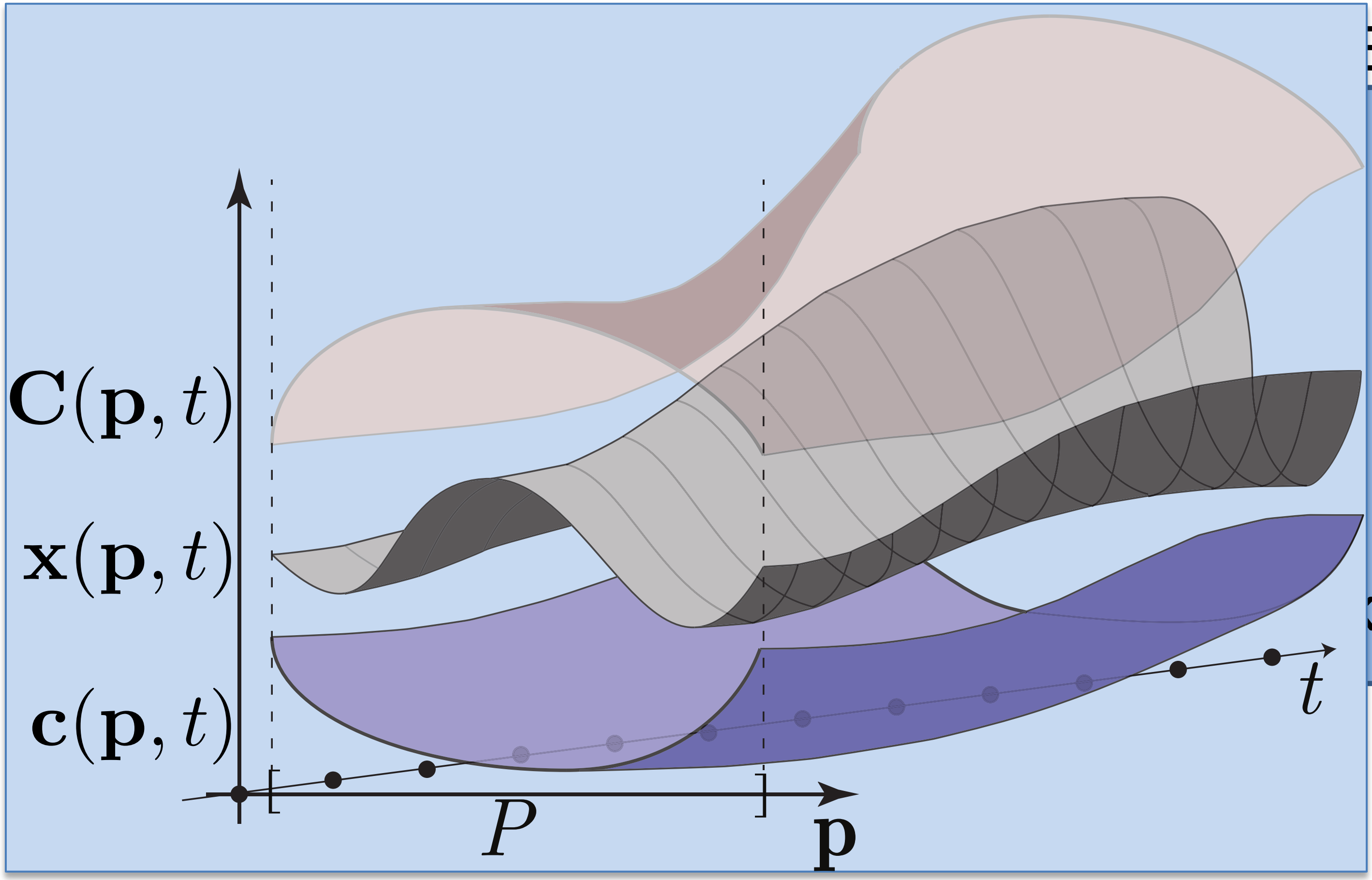
Global optimization problems constrained by ODEs/DAEs:

$$\dot{\mathbf{x}}(\mathbf{p},t) = \mathbf{f}(\mathbf{x}(\mathbf{p},t),\mathbf{p},t), \forall t \in I = [t_0, t_f]$$
$$\mathbf{x}(\mathbf{p},t_0) = \mathbf{x}_0(\mathbf{p})$$

$$\dot{\mathbf{c}}(\mathbf{p},t) = \mathbf{u}(\mathbf{p},\mathbf{c}(\mathbf{p},t),t),$$
$$\mathbf{c}(\mathbf{p},t_0) = \mathbf{c}_0(\mathbf{p}),$$
$$\dot{\mathbf{C}}(\mathbf{p},t) = \mathbf{o}(\mathbf{p},\mathbf{C}(\mathbf{p},t),t),$$
$$\mathbf{C}(\mathbf{p},t_0) = \mathbf{C}_0(\mathbf{p})$$

Relax-then-Discretize (McCormick relaxations)

Recent Advances: ODEs/DAEs



Es/DAEs:

$$\dot{\mathbf{c}}(\mathbf{p},t) = \mathbf{u}(\mathbf{p},\mathbf{c}(\mathbf{p},t),t),$$
$$\mathbf{c}(\mathbf{p},t_0) = \mathbf{c}_0(\mathbf{p}),$$
$$\dot{\mathbf{C}}(\mathbf{p},t) = \mathbf{o}(\mathbf{p},\mathbf{C}(\mathbf{p},t),t),$$
$$\mathbf{C}(\mathbf{p},t_0) = \mathbf{C}_0(\mathbf{p})$$

relaxations)

Recent Advances: ODEs/DAEs

SourceCodeMcCormick.jl

$$\frac{dx_i}{dt} = x_i + p_i \longrightarrow \begin{array}{l} \dfrac{dx_i^L}{dt} = x_i^L + p_i^L, \quad \dfrac{dx_i^U}{dt} = x_i^U + p_i^U, \\[2ex] \dfrac{dx_i^{cv}}{dt} = x_i^{cv} + p_i^{cv}, \quad \dfrac{dx_i^{cc}}{dt} = x_i^{cc} + p_i^{cc} \end{array}$$

Source code transformation to generate the auxiliary ODE system

Utilizes ModelingToolkit.jl's ODE format, with numerical integration through the SciML ecosystem

Recent Advances: ODEs/DAEs

SourceCodeMcCormick.jl

```
using SourceCodeMcCormick, ModelingToolkit

@parameters p[1:2] t
@variables x[1:2](t)
D = Differential(t)

eqns = [D(x[1]) ~ p[1]+x[1],
        D(x[2]) ~ p[2]+x[2]]

@named syst = ODESystem(eqns, t, x, p)
new_syst = apply_transform(McCormickIntervalTransform(), syst)
```

$^U_i + p_i^U,$

$x_i^{cc} + p_i^{cc}$

Source code tra... m

Utilizes ModelingToolkit.jl's ODE format, with numerical integration through the SciML ecosystem

Recent Advances: ODEs/DAEs

SourceCodeMcCormick.jl

```
Original system:
Differential(t)(x[1](t)) ~ x[1](t) + p[1]
Differential(t)(x[2](t)) ~ x[2](t) + p[2]

Transformed system:
Differential(t)(x_1_lo(t)) ~ p_1_lo + x_1_lo(t)
Differential(t)(x_1_hi(t)) ~ p_1_hi + x_1_hi(t)
Differential(t)(x_1_cv(t)) ~ p_1_cv + x_1_cv(t)
Differential(t)(x_1_cc(t)) ~ p_1_cc + x_1_cc(t)
Differential(t)(x_2_lo(t)) ~ p_2_lo + x_2_lo(t)
Differential(t)(x_2_hi(t)) ~ p_2_hi + x_2_hi(t)
Differential(t)(x_2_cv(t)) ~ p_2_cv + x_2_cv(t)
Differential(t)(x_2_cc(t)) ~ p_2_cc + x_2_cc(t)
```

$$\frac{dx_i^L}{dt} = x_i^L + p_i^L, \quad \frac{dx_i^U}{dt} = x_i^U + p_i^U,$$

$$\frac{dx_i^{cv}}{dt} = x_i^{cv} + p_i^{cv}, \quad \frac{dx_i^{cc}}{dt} = x_i^{cc} + p_i^{cc}$$

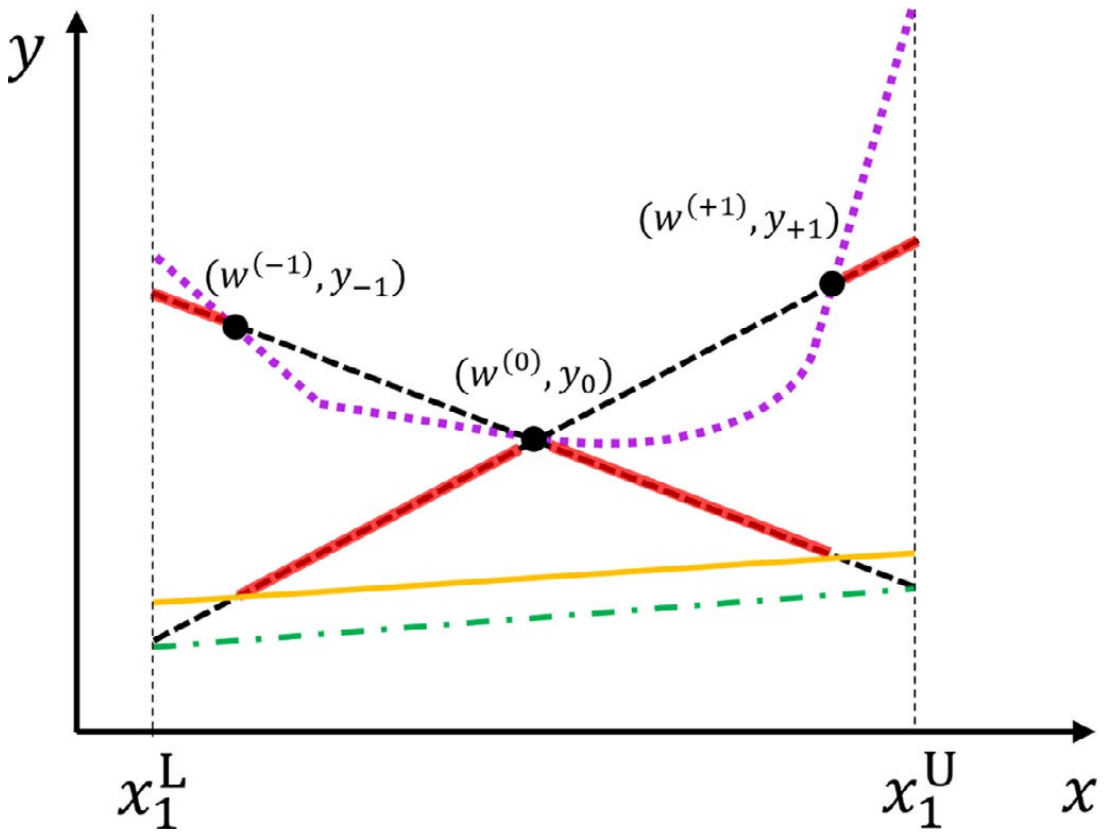
S... e the auxiliary ODE system

Utilizes ModelingToolkit.jl's ODE format, with numerical integration through the SciML ecosystem

Recent Advances: ODEs/DAEs

Lower-bounding problem solved using black-box sampling method published by Song et al.[1]

Lower bound constructed using pointwise evaluations

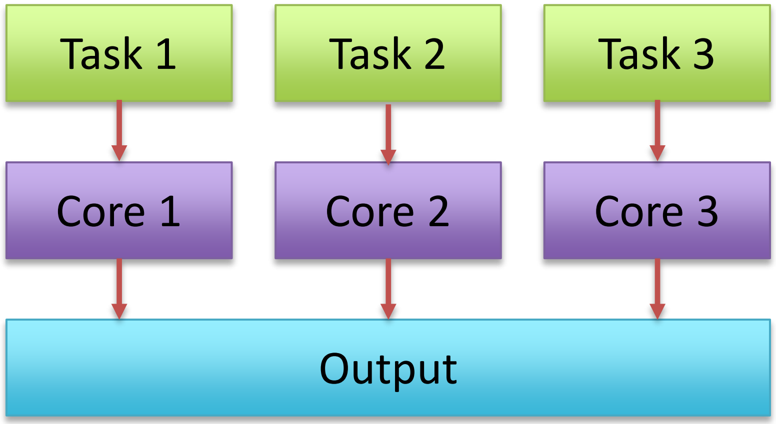


1. Song et al. Bounding convex relaxations of process models from below by tractable black-box sampling(2021)

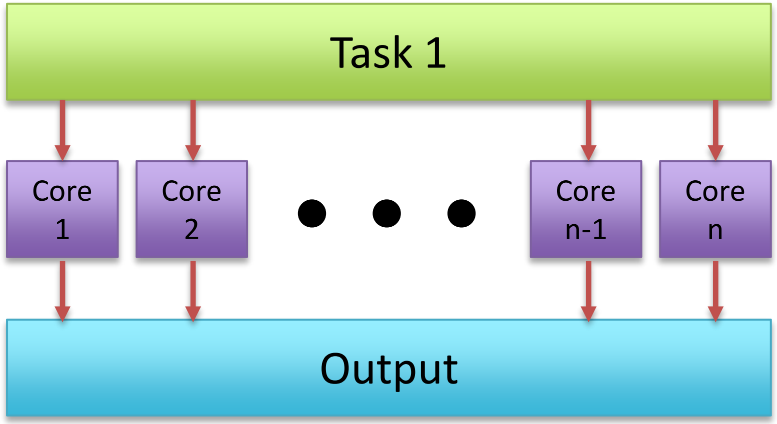
Recent Advances: ODEs/DAEs

Hardware solution: exploit GPGPU architecture for ODE parallelization

CPU Workflow (x86 cores)

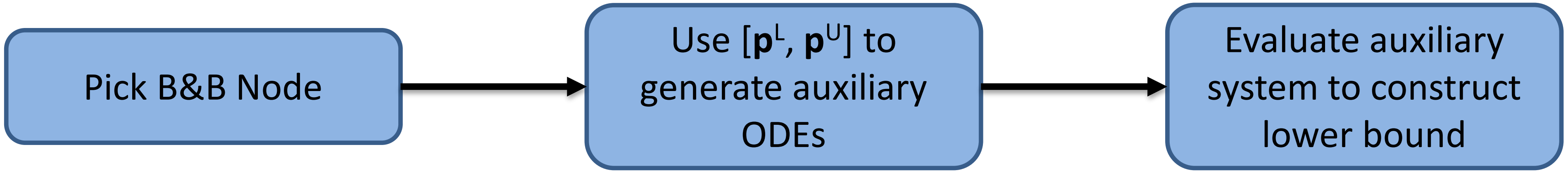


GPGPU Workflow (CUDA cores)

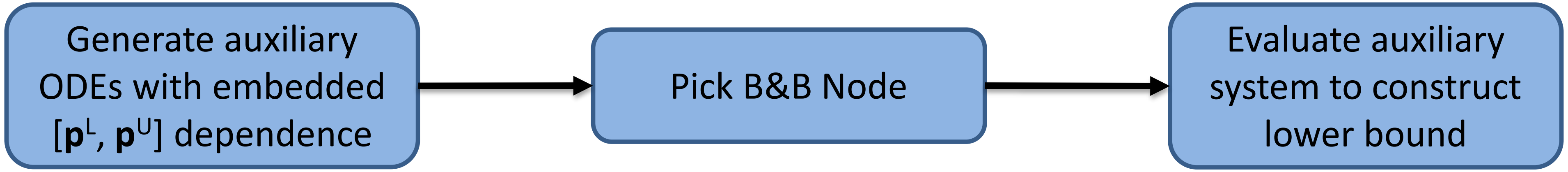


Recent Advances: ODEs/DAEs

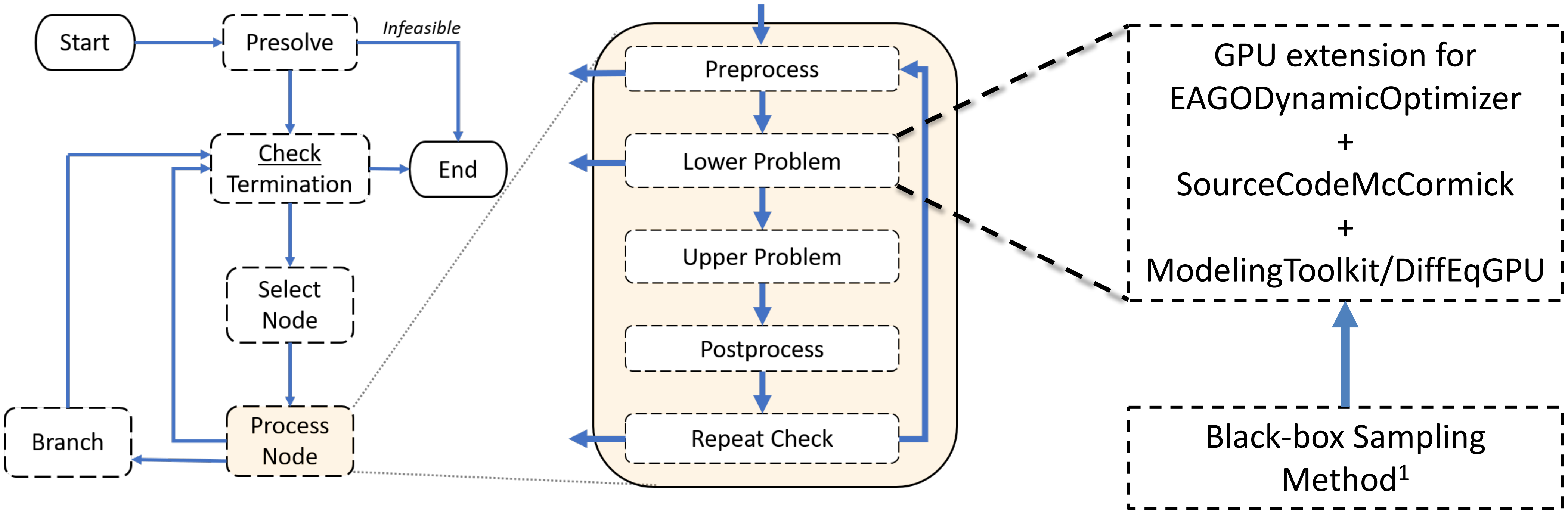
Typical relax-then-discretize



SourceCodeMcCormick.jl relax-then-discretize

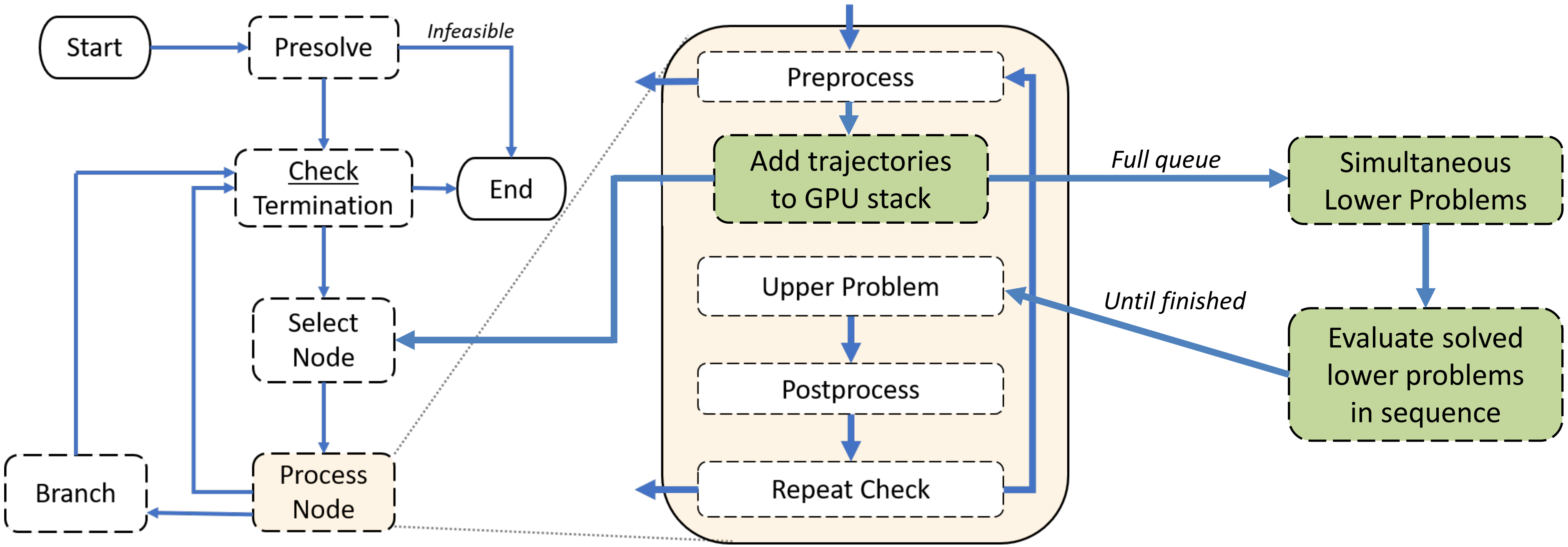


Recent Advances: ODEs/DAEs



[1]Song et al. paper

Recent Advances: ODEs/DAEs



Conclusion

- EAGO is a feature-rich nonconvex solver and research platform written in Julia
  - Use with JuMP AML or stand-alone
- Exhaustive library of relaxation envelopes
  - New theory for composite bilinear relaxations
  - ANNs
- Currently focusing on relaxations for global dynamic optimization
  - Source code transformation
  - Exploiting GPGPU

Thank You – Any Questions?

- PSORLab@UCONN
- EURO 2022 Organizers
- Funding: National Science Foundation

This material is based upon work supported by the National Science Foundation under Grant No. 1932723. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

https://www.psor.uconn.edu

https://www.github.com/PSORLab/EAGO.jl

UCONN
UNIVERSITY OF CONNECTICUT

3-6 JULY
EURO 2022
AALTO UNIVERSITY
ESPOO FINLAND

Process Systems and Operations Research Laboratory